

Determining the Necessity of Fault Tolerance Techniques in FPGA Devices for Space Missions

Louis Daniël van Harten^a, Mahsa Mousavi^a, Roel Jordans^{a,b,*}, Hamid Reza Pourshaghghi^{b,a}

^a*Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands*

^b*Radboud Radio Lab, Department of Astrophysics/IMAPP, Radboud University, Nijmegen, The Netherlands*

Abstract

Functionality of electronic components in space is strongly influenced by the impact of radiation induced errors which may interfere with the proper operation of the equipment. In space missions, FPGA implementations are generally protected using computationally expensive radiation-error mitigation techniques such as error correcting codes (ECC) and triple modular redundancy (TMR). For high-performance systems, such fault tolerance techniques can prove problematic due to both the added computational requirements and their resulting power overhead. As such it is important to make a proper assessment of the expected error rates to make a proper selection of mitigation techniques.

This paper provides an extensive overview of the techniques used for determining the necessity of such mitigation techniques in space missions and other situations where a large radiation dose will be encountered. Given the presented study and radiation analysis, in this paper an experimental example is presented in the form of a case study on the Digital Receiver System (DRS) in the Netherlands-China Low-frequency Explorer (NCLE) mission, which is implemented using a Xilinx Kintex-7 SRAM FPGA. Fault rates are estimated for a five-year mission to the second Earth-Moon Lagrange point (L2) and the chosen fault mitigation strategy as implemented in NCLE-DRS is presented.

The effect of potential upsets on the functionality of DRS has been taken into account in order to make error estimations more precise. Thus, two test-benches are developed and presented to experimentally evaluate the effect of upsets in FPGA configuration memory and the data on the DRS final outputs.

The approach provided in this paper should generalize well to other space missions, as long as a general estimate of the expected radiation environment is available.

*Corresponding author

Email address: `r.jordans@tue.nl` (Roel Jordans)

1. Introduction

Operating electronics in space brings several challenges, one of which is the effects of radiation induced faults. This paper presents a workflow for determining whether implementing classical fault tolerance techniques is necessary for the successful completion of a mission, guided by a case study on a lunar mission.

The case study discussed in this paper targets the Netherlands-China Low-frequency Explorer (NCLE), a low-frequency radio instrument previously selected for the 2018 Chinese Change 4 mission to an Earth-Moon L2 halo orbit¹. The NCLE instrument consists of three 5-meter long monopole antennas that are mounted onto the relay satellite (called Queqiao), together with the related data acquisition and processing electronics. It has a nominal mission lifetime of at least 3 years and it is designed to be sensitive in the 80 kHz to 80 MHz frequency range. Low-Frequency radio astronomy (below 30 MHz), can only be done well from space due to effects such as the cut-off in the Earth's ionosphere, the man-made Radio Frequency Interference (RFI), the Earth's Auroral Kilometric Radiation (AKR) and Quasi-Thermal Noise (QTN) that make sensitive measurement from ground-based facilities impossible [1, 2, 3]. NCLE was launched successfully on May 21st from Xichang launch center in China.

The data processing part of the NCLE instrument is the Digital Receiver System (DRS). This DRS, implemented using a Xilinx Kintex-7 SRAM FPGA, is tasked with processing and storing the data obtained from the three monopole antennas from which the radio signals are sampled at 120MHz. The on-board data processing entails multiple parallel Fourier transforms of up to 16k points each, as well as digital band selection, filtering, sub-sampling, and data correlation and accumulation operations. All of these operations need to be performed in real-time on the input data streams from the antennas and within a power budget of less than 3 Watts on average for the FPGA. The reason for this constraint is not a shortage of power on the supply side; the limiting factor is the amount of heat that can be dissipated safely.

The DRS implements several modes in a flexible software-defined radio system. These modes for instance perform Fast Fourier Transforms (FFTs) to create average radio spectra in different frequency bands with different frequency resolutions. In this paper, the term *observation mode* will therefore refers to an application running on the FPGA.

Previous investigations of the Xilinx Kintex 7 FPGA have shown that the architecture is susceptible to radiation-induced upsets [4, 5, 6, 7], which may introduce errors in the computed results. In our analysis we determine which (if any) fault tolerance techniques, such as triple modular redundancy (TMR) or error correcting codes (ECC)[8], are needed to reduce these faults in the NCLE

¹Halo orbit lies approximately 64500km beyond the moon, which allows satellite for communication with equipment landing on the moon surface on the backside of the moon such as the companion lander to the Chang'e 4 mission.

DRS to acceptable levels. However, blindly applying these techniques incurs significant overhead, dramatically increasing both the required FPGA resources and the power consumption. For example, TMR increases both the required resources and power by approximately a factor of three. Overhead in this order of magnitude would undermine a part of the desired science goals, as smaller-than-preferred Fourier transformations would have to be used, reducing the spectral resolution and overall value of the gathered astrophysical observation data.

We introduce a method that is used to determine the susceptibility of the system to radiation induced faults. This method consists of three main steps: environmental analysis, statistical analysis and an experimental part. In the environmental analysis step, required information on the radiation environment at the mission destination is gathered. In the statistical step, the previous analysis is combined with device utilization statistics from our FPGA synthesis flow together with the cross-section metrics for various resource types on the Kintex-7 FPGA obtained from literature. As the cross-section reported in literature is not application-specific, both the FPGA application architecture and behavior still need to be taken into account. The overall application behavior and implementation choices can cause individual upsets to be masked such that these upsets do not affect the functionality of applications. Thus, in the final (experimental) step, we implement two test-benches to calculate the effective cross-section of a representative design by injecting upsets into the design and to evaluate the design functionality in the presence of the expected upsets.

This analysis provides an insight into the expected error rate in various FPGA components and the propagation of faults, which then allows us to determine whether it would be a viable approach to implement the system with only limited application of fault tolerance techniques. This approach is described in this paper in detail along with several comments on how this analysis could be adapted to different use-cases.

The work presented here is an extension of [9]. This extension includes improved background explanations, updated results based on additional Design Vulnerability Factor (DVF) analysis and measurements, and accordingly updated final results. The goal of this work is to present the rationale behind, and selected approach for, the radiation-resistant design of the NCLE Digital Receiver System. This work is aimed to enable other teams to replicate the presented analysis with their respective (space based) use-cases.

This paper is organized as follows. Section 2 gives an overview of the logical lay-out and physical situation of the NCLE payload in the Chang'e 4 relay satellite. Section 3 discusses simulation results of the radiation environment at the mission destination, in order to provide an estimate for the expected radiation sustained by the system over the mission duration. After that, Section 4 gives an overview of the susceptibility of the system to various types of radiation errors and degenerative effects, as well as their impact on the system. Where applicable, vulnerable cross-sections are determined. In addition, this section describes the analysis of the DRS processing pipeline, estimating the percent-

age of data-upsets that propagate to the system output. Section 5 provides an explanation how the critical cross-section of a design can be determined. It also quantifies two definitions of the critical cross-section of the FPGA implementation: faults that produce errors in the observed data, as well as the subset of those errors which have a significant and possibly catastrophic effect on the science data, together with their respective estimated incident rates. Finally, Section 6 presents the selected mitigation techniques and the decision process based on the previously obtained metrics.

2. System setup

The engineering model of the NCLE Digital Receiver System module is pictured in Figure 1. In this picture, the longer board at the bottom is the data processing board containing the Xilinx XC7K160T FPGA which is the focus of this study. This board contains the data flash memory modules used for storing acquired observation data as well. The shorter board at the top enclosed in the aluminium casing contains the ADC module. This module has three ADC channels servicing the three antennas on the NCLE, and a fourth ADC channel that serves as input from the calibration source. In the back of the picture both boards are connected using a rigid backplane structure, which also provides connectivity to the command and data handling system (CDHS) of the instrument. This backplane provides a command and control interface, a data download interface, and an access channel to the FPGA configuration flash, making it possible to track housekeeping data and push updates to the FPGA design. The CDHS is tasked with communicating gathered science data back to Earth through the Chang’e 4 satellite communication system, and it also acts as a monitor for the DRS FPGA. Finally, the CDHS monitors the power consumption and receives a heartbeat signal from the DRS board and can restart and reconfigure the FPGA when unexpected behaviour is observed, for example, after a non-recoverable error has been observed within the FPGA.

During its operation the FPGA will be processing sample data from the ADC board, 14bit samples arriving at 120MSPS for each of the three antenna signals, together good for 5Gb/s of input data. These samples are then passed through a filter bank for optional sub-band selection and processed with a 16k point FFT. After that the signals of the three antennas are correlated and accumulated before sending them to the flash storage. Overall these computations take a significant part of the processing resources of the available FPGA and limit the possibilities for using TMR in the design. Furthermore, these computations result in a significant power draw on the FPGA, up to 3W can be spent on these computations depending on the filter bank and accumulation settings. With the DRS board operating in a vacuum environment this requires extra attention for the cooling system, even without including redundant computations. A passive heat-coupling system has been designed for transporting the thermal energy towards the satellite body. This cooling system is expected to allow the FPGA to operate at reasonably safe temperatures within the ex-

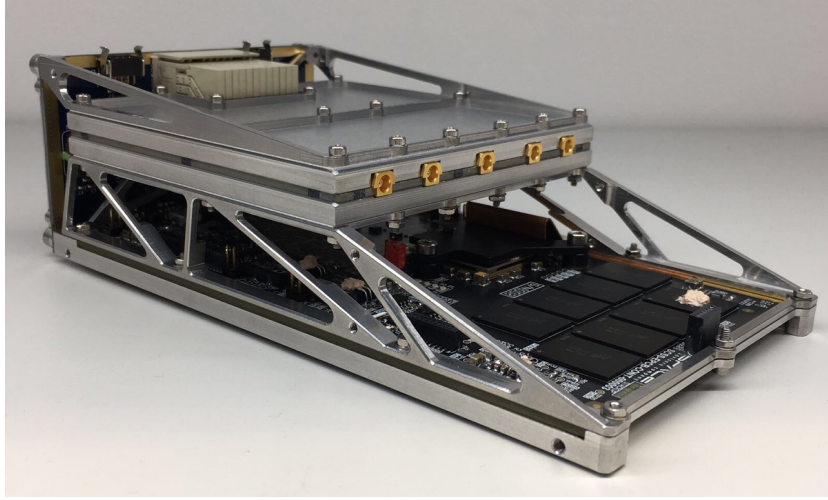


Figure 1: The engineering model of the NCLE Digital Receiver System during FPGA firmware development. The FPGA board assembled together with its mounting bracket, the ADC module, and the interconnecting backplane.

pected mission environment. A much more complicated liquid cooling system is likely to be required to support higher power consumption levels within the FPGA. Overall, thermal considerations and the limited FPGA resources available when implementing the full science processing requirements required the in-depth analysis of the expected fault rates as only limited resources for error correction are available.

Analyzing the effects of radiation requires a good estimate on the amount of radiation reaching the FPGA die. This radiation strength depends on both the environmental situation and the amount of shielding provided as part of the electronics box and satellite frame. While working on this modeling, the shielding thickness was still unknown. Due to weight budget limitations however we had to assume only a relatively light shielding. When no further justification is available, the ESA standard recommends assuming 1 g/cm^2 of shielding as suggested by the ESA space environment engineering standard [10], which is equivalent to approximately 3.7mm of aluminium in thickness. However, as the mechanical housing still had to be designed within the project, a more conservative estimate of 2mm of aluminium shielding was used for the analysis. As a result, the fault rates presented in this paper are assumed to be pessimistic and should provide a safe estimate for our mission profile.

3. Radiation environment at mission destination

The first step in estimating the radiation induced errors is to obtain an overview of the radiation environment. Ideally, both the radiation dose and the

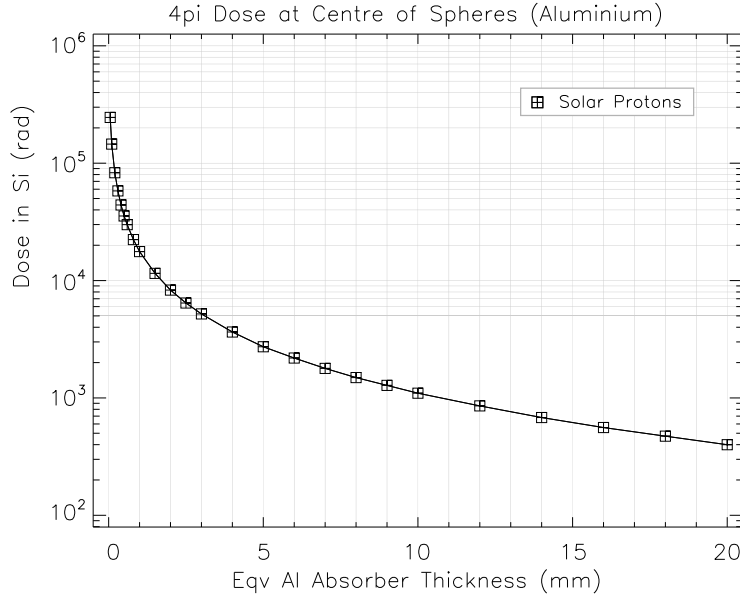


Figure 2: Five-year ionizing dose in near Earth interplanetary orbit at 1.0 AU from the sun, starting from March 2018. Results obtained from SHIELDOSE simulations via SPENVIS.

spectrum shape of particle energies should be known, along with the type of expected particles. The latter two determine the effective linear energy transfer (LET) of the particles to the device. The effective energy transfer of a particle impact is (approximately linearly) related to upset rate [6, Fig. 1]. By combining the spectrum of expected LETs with the effective cross-section of the device (faults per amount of flux), the expected number of faults can be obtained.

Estimations for space environmental influences on satellites can be obtained from ESA’s SPace ENVironment Information System (SPENVIS) [11]. This system contains radiation models for several locales within our solar system, taking into account the solar cycle, background radiation, influences of a nearby planet², and space debris such as micro-meteorites. For the Earth-Moon L2 Lagrange point, the SPENVIS documentation suggests to use a ‘deep space’ model and selecting a solar orbit at 1AU, as it is distant enough from the Earth to be beyond its sphere of influence. For our mission, we selected a five-year flight launching in May 2018. These estimates only focus on the mission lifetime during its deployment around the Earth-Moon L2 point and exclude any radiation effects captured during the travel to that point. A more detailed modeling is also possible in SPENVIS by providing multiple parts of the spacecraft trajectory. In our case the instrument will only be powered during its deployment phase, as such, adding the other parts of the trajectory to the computations

²SPENVIS currently has models for the effects of Mercury, Earth, Mars, and Jupiter.

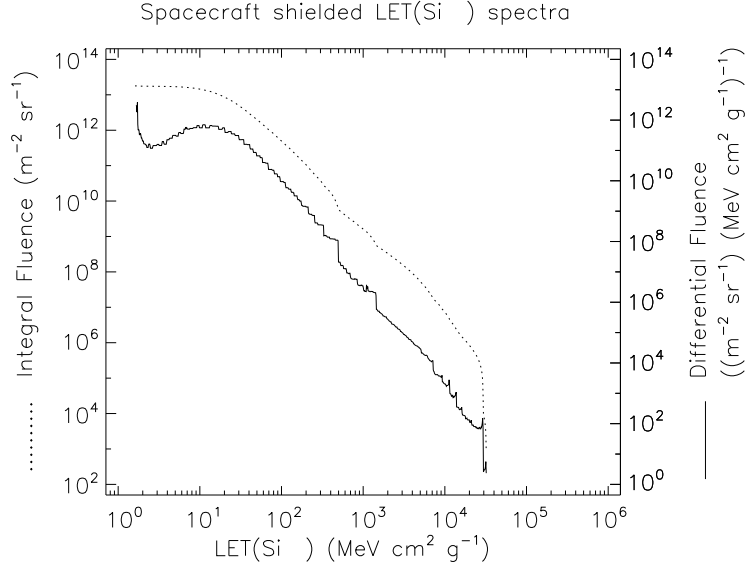


Figure 3: Simulated five-year fluence plotted as LET spectrum, 2mm of Al shielding.

would reduce the accuracy of our expected fault rates.

3.1. Total dose estimates

The total ionization dose (TID) impacts the expected lifetime of electronic components. The modeling of our spacecraft trajectory as a 1AU solar orbit in SPENVIS ignores the intermittent shielding effect of the Moon and the Earth, as well as the effects of intermittently passing through the Earth's magnetotail, but should still provide a reasonable estimate.

From SPENVIS, the SHIELDOSE-2Q [12] simulation was run for a spherical aluminium shield around a silicon target to estimate the total dose. Results from this simulation can be found in Figure 2. We find an estimated TID of approximately 8krad(Si) for 2mm of shielding over the extended five-year mission lifetime.

3.2. Particle fluence estimates

Also using SPENVIS, a prediction for the long-term LET spectrum was obtained, with total fluence as a function of particle energy transfer. These results can be found in Figure 3. Note that the unit for integral fluence in this figure is $m^{-2}sr^{-1}$; particles per area, per steradian. The simulation assumes an isotropic radiation source and calculates flux through a spherical shield. To convert these results to particles hitting a flat surface of a certain area, it is necessary to calculate the projection of the flat chip area to a sphere. Integrating over both the azimuth and elevation, this boils down to equation

1, i.e., the results should be multiplied by 2π in order to get the amount of particles per square meter per second.

$$\int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} A |\cos(\theta) \sin(\theta)| d\phi d\theta = 2\pi A \quad (1)$$

For the NCLE mission, this results in a total integral fluence of $1.1 \cdot 10^{14} m^{-2}$ over the mission lifetime, equivalent to an average flux of $67 cm^{-2} s^{-1}$.

3.3. Worst case flux

Due to the nature of varying solar conditions during the solar cycle and especially active periods such as solar flares, average particle flux can differ wildly from the worst case scenario. In additional simulations, results for various worst case scenarios were obtained from SPENVIS. The obtained results are shown in Table 1.

Table 1: Expected particle flux as predicted using SPENVIS

Situation	Flux ($cm^{-2} s^{-1}$)
average	$6.7 \cdot 10^1$
worst week	$6.4 \cdot 10^3$
worst day	$2.9 \cdot 10^4$
worst 5 minutes	$1.1 \cdot 10^5$

An important observation is that during the worst five minutes, the expected flux is approximately 1600 times higher than in the average case. In addition, this implies that in the median situation, the flux is likely to be significantly lower than in the average case. In our error rate estimation we will perform the analysis for the average particle flux and the particle flux during each of the provided worst-case periods.

4. Overview of radiation errors

Radiation errors in FPGAs come in several categories: Configurable Logic Block (CLB) errors, BRAM upsets, configuration (SRAM) upsets, destructive latch-ups and total ionization dose failure. The following sections will go over each of these, listing the effects, expected incident rate, and (if needed) propose defensive strategies.

For single event effects, the effective cross-section of the system is discussed. This is defined as the amount of events per unit of fluence. It combines the probability that a particle impact will cause an upset with the amount of particles passing through the chip. The quantity of fluence is the inverse of area, which makes area per bit (or area per system) the quantity for the cross-section. A meaningful interpretation of this concept is the critical area which a particle has to strike in order to cause an event.

4.1. Destructive latch-ups

When a heavy ion strikes a silicon CMOS microcircuit, there is a chance for a latch-up to occur: a self-sustaining parasitic short-circuit, which draws high current and may break the circuit due to high temperatures. Surprisingly, the Kintex-7 does not seem to suffer from the same destructive latch-ups found in many other FPGAs [6, 7].

Heavy ion testing in the TAMU K500 Cyclotron facility [6] has shown that latch-ups in the Kintex-7 only draw approximately 125 mA from the VCCAUX line (1.8V, meant for auxiliary circuits such as clock managers and dedicated configuration pins), which is not enough to cause any lasting damage to the circuit. The exact cause for the draw is not clear. In their initial tests, the FPGA was operated above its normal operating voltage in order to trigger and study the latch-up behavior. Additional testing in the same facility at nominal voltages has shown that the event only occurs for very high energy particles; the lowest effective LET at which this phenomenon was observed is $1.5 \cdot 10^4 \text{ MeV cm}^2/\text{g}$, at which the estimated cross-section was determined as approximately $5 \cdot 10^{-7} \text{ cm}^2$. These results were confirmed by heavy-ion tests at the Cyclotron Resource Center in Louvain [7], where similar behaviour was observed with a threshold SEL of $1.56 \cdot 10^4 \text{ MeV cm}^2/\text{g}$. The expected five-year integral fluence in L2 for events of at least this LET is only approximately $1.0 \cdot 10^5 \text{ cm}^{-2}$, meaning the chance that a single event of this type occurs during the five-year mission is approximately 5%. Results for additional test are available [5], in which a Kintex-7 device is irradiated with $1.9 \cdot 10^{11} \text{ cm}^{-2}$ fluence of 105 MeV protons. In this test, not a single latch-up was detected.

An additional note from the Cyclotron Louvain tests is that the latch-ups in the Kintex-7 do not seem to cause any loss of part functionality. Power cycling the device removes all symptoms. These results imply that for Kintex-7 applications in space, latch-ups do not pose a threat in the form of lasting damage to the FPGA. It can be concluded that because of the extremely low incident rate and low impact on the system, the effects of latch-ups in the Kintex-7 can effectively be ignored.

4.2. Total Ionizing Dose effects

The total ionizing dose (TID) effects mostly consist of the transistors in the FPGA slowly breaking down by particles hitting the doped silicon and slowly weakening the doping. This results in the transistors slowing down (and eventually breaking), resulting in a longer delay in the critical path.

Few elaborate tests have been published researching the TID effects in the Kintex-7 specifically. However, these test have been performed on other FPGA devices, such as on the Lattice ECP3 [13, Fig. 4]. It is likely reasonable to assume the results for the Kintex-7 would follow a similar pattern: negligible slowdown up to approximately halfway the device failure point, after which the slowdown gradually increases up to the point of total device failure. The Lattice ECP3, of which test results were mentioned above, is an FPGA which uses 65nm

technology [14]. The Kintex-7 used in the NCLE project uses 28nm technology. Perhaps unintuitively so, smaller nanometer technologies are generally less prone to TID transistor slowdown effects due to their smaller oxide thickness [15]. This means that the Kintex-7 is unlikely to perform worse than the LFE3-35EA under similar large ionizing dose conditions.

There is some data available on Kintex-7 TID failure points and all data points seem to confirm the above mentioned assumption. One paper presented a Kintex-7 FPGA being irradiated with 105 MeV protons for a total dose of 17.0 krad [5] (170 Gy). No functional problems were observed. In another experiment, two Kintex-7 devices were irradiated in an attempt at finding the device failure point [16]. The first broke down after receiving 340 krad (3400 Gy) and one other still functioned after receiving 446 krad (4460 Gy), after which the test was aborted. Both tests were performed using high energy (180 MeV) protons. As shown in Section 3.1, the expected total mission dose is less than 10 krad (100 Gy): this is more than 30 times lower than the lowest observed failure point of a Kintex-7. Considering these results, it is reasonable to assume TID effects in the FPGA can be ignored in the NCLE mission.

4.3. Configuration upsets

Single event upsets (that is: an incidental flip in the state of an element) can be especially troublesome in an FPGA, as upsets can affect the state of configuration bits. This means the logic as composed by the digital gates functionally changes. Depending on the bit that was struck, this can lead to faulty data output or bring the system into an erroneous state.

4.3.1. Expected rate of configuration upsets

Providing a reliable estimate on the amount of expected faults is not entirely trivial. While there are many papers available in which relevant radiation test results are presented, the type and amount of radiation is different from the expected radiation at the L2 point, to a partially unknown degree.

Test results from literature have shown that for a Kintex-7 device in a 105 MeV proton beam, the effective configuration upset cross-section is $5.21 \cdot 10^{-15} \text{ cm}^2/\text{bit}$ [5]. This proton beam translates to a LET spectrum where approximately 40% of the events have a LET of at least $1 \text{ MeVcm}^2/\text{mg}$ and 10% has a LET of at least $8 \text{ MeVcm}^2/\text{mg}$ (similar to the one shown in [17, Fig. 2]). Comparing this to the expected mission spectrum shown in Fig. 3, it is clear that the proton tests are not entirely representative of the expected radiation environment. In the expected mission spectrum, only one in ten thousand events have a LET of at least $1 \text{ MeVcm}^2/\text{mg}$ and less than one in a million events have a LET of at least $8 \text{ MeVcm}^2/\text{mg}$.

Literature has shown that for high LET ($5 - 20 \text{ MeVcm}^2/\text{mg}$), the configuration cross-section is reasonably linear with the event energy [6]. However, extrapolating these results to low energy events might not be valid; there is insufficient evidence supporting this relation behaves the same way at low LET.

Table 2: Expected configuration upsets and uncorrectable MBUs

Situation	Configuration upsets	Uncorrectable upsets
average	0.63 / day	$3.2 \cdot 10^{-4}$ / day
worst week	60.4 / day	0.030 / day
worst day	274 / day	0.14 / day
worst 5 minutes	1038 / day	0.52 / day

Applying the 105 MeV proton cross-section as the cross-section average would mean ignoring the discrepancy between the beam spectrum and the expected mission spectrum. This is equivalent to pretending the system impact of particles hitting the device is larger on average than can reasonably be assumed. This will result in inflated error estimates; while unfortunate, it is better to stay on the side of “unrealistic worst-case” than to end up with an estimate that is significantly too low. As such, in this paper, the cross-sections measured using high-energy protons will be considered as valid.

In the concept design of the NCLE digital receiver system FPGA implementation, approximately 20Mbit of configuration SRAM is used. This translates to an effective configuration cross-section of $1.1 \cdot 10^{-7} \text{ cm}^2$. Combining this result with the flux from Section 3 results in the average upset-rates found in Table 2. Note that the numbers for the worst five minutes are extrapolated beyond the five-minute duration for consistency in the table.

4.3.2. Configuration scrubbing

The main technique to combat configuration upsets is “scrubbing”. This means to constantly compare the active configuration bits with a protected (duplicated) reference and to reconfigure blocks where necessary, or by achieving similar functionality using error correcting code (ECC) bits. The Kintex 7 used in the NCLE has an on-board configuration scrubber, which is able to correct single-bit errors in one word, and detect up to two-bit errors. Additionally, a more advanced single-error mitigation IP core is available, which is able to correct up to two-bit adjacent errors in one word, and detect any larger odd-count bit errors in one word, as well as some larger even-bit errors [18]. In further analysis, it is assumed this core is used. Advanced two-bit adjacent error correction relies on storing additional Cyclic Redundancy Check (CRC) bits in the device BRAM. For designs that, unlike the NCLE DRS, use (almost) all BRAM tiles in the FPGA, this option is not available. In such cases, all multi-bit upsets should be considered uncorrectable.

It is also important to note that the scrubber does not fix upset bits immediately. The SEM core has a scrubbing latency of 12.9 ms for the specific FPGA in the NCLE [18], although this can be increased to save power. There is also a correction latency of 0.6 ms. This means that whenever a correctable upset occurs, the system is stuck in an imperfect state for up to approximately fifteen milliseconds.

Table 3: Estimated resource utilization

Resource	Utilization (% of available)
CLB slices	11250 (44%)
BRAM tiles (36k)	210 (65%)
DSP blocks	400 (67%)

4.3.3. Multi-bit upsets and correctability

The earlier mentioned configuration scrubber in the Kintex-7 can correct up to two-bit errors, and detect all odd-number bit errors in a single word. While the bits of different words are physically interleaved to combat multi-bit upsets in a single word, these do happen occasionally. In literature, an upset in multiple bits across different words is sometimes called an MCU (multi-cell upset), whereas the term MBU (multi-bit upset) is reserved for upsets which flip multiple bits in a single word.

Fairly extensive testing has been done in prior works to characterize the multi-bit upset behavior in the Kintex-7 [19]. The tests with the lowest energy ions used nitrogen and oxygen ions with an energy of 200 MeV , impacting with an average LET of $1.16\text{ MeVcm}^2/\text{mg}$ and $1.54\text{ MeVcm}^2/\text{mg}$ respectively. Out of the presented tests, these should be most representative of the expected mission environment. All but one in ten thousand particle strikes during the mission is expected to have a LET of less than the average of the ion strikes in these tests, meaning the test results convey an absolute worst case.

In the nitrogen and oxygen ion tests, average incident rates of 0.4% and 0.05% were found for 2-bit adjacent MBUs and 3-bit MBUs respectively, as a fraction of all configuration upsets. The amount of ≥ 4 -bit MBUs and non-adjacent 2-bit MBUs was negligible. The average incident rate of MCUs was found to be 1.7%. These rates result in a total uncorrectable configuration cross-section of $5.5 \cdot 10^{-11}\text{ cm}^2$ and a negligible undetectable cross-section. The expected rate of uncorrectable upsets during the mission can be found in Table 2.

4.4. Data upsets

Apart from configuration upsets, there are several other possible single event faults: errors in BRAM, propagated transients in multipliers and upsets in flip-flops. These errors are all related to user-data and it is not possible to detect them, unless logic is specially generated for that purpose.

As with the used configuration memory, the exact utilization of the BRAM-blocks, DSP-blocks and flip-flops in the final design is not yet fully known. Estimates were made from the concept design, which can be found in Table 3. Non-listed sites (such as distributed RAM blocks and Muxes) have a sufficiently low expected usage that their cross-section contribution was deemed insignificant. These estimates are effectively the resource utilization of the concept

Table 4: Cross-sections of various slices/blocks

Resource	Cross-section per unit	In concept design
CLB slices	$4.22 \cdot 10^{-14} \text{ cm}^2/\text{slice}$	$0.47 \cdot 10^{-9} \text{ cm}^2$
BRAM tiles	$4.81 \cdot 10^{-11} \text{ cm}^2/\text{tile}$	$10.1 \cdot 10^{-9} \text{ cm}^2$
DSP blocks	$9.88 \cdot 10^{-13} \text{ cm}^2/\text{block}$	$0.40 \cdot 10^{-9} \text{ cm}^2$

Table 5: Expected rate of data upsets

Situation	Data upsets per day
average	0.064
worst week	6.07
worst day	27.5
worst 5 minutes	104.3

design, rounded up slightly.

Cross-sections in a proton beam for BRAM upsets, DSP blocks and CLB slices (which contain LUTs and Flip Flops) are available from literature, obtained in similar conditions as the configuration memory cross-section results used in Section 4.3.1. The cross-sections for BRAM upsets, DSP blocks, and logic slices for a Kintex-7 7K325T were determined in literature as $2.17 \cdot 10^{-9} \text{ cm}^2/\text{device}$ (logic slices), $0.83 \cdot 10^{-9} \text{ cm}^2/\text{device}$ (DSP blocks) and $21.4 \cdot 10^{-9} \text{ cm}^2/\text{device}$ (BRAM) for full utilization of those respective parts [5]. The device under test in this paper was a larger FPGA than the 7K160T in the NCLE system, so some conversion is necessary.

Results from converting the cross-sections to per unit and full design cross-sections are shown in Table 4. Note that while it provides a convenient intermediate step in calculating the design cross-section from device cross-sections, converting the per device cross-sections to per logical slice cross-sections is somewhat fictitious, as many of these components are not truly separate blocks on the FPGA. Adding all of these cross-sections together results in a total user data cross-section of $1.09 \cdot 10^{-8} \text{ cm}^2$, which happens to be almost exactly 10% of the effective configuration cross-section in this case. The resulting expected upset rate in the user data for various situations is given in Table 5.

5. Design Sensitivity and the Critical Cross-section

The upset rates presented in the previous section are only based on the resource usage of various elements in the FPGA and their physical characterizations. However, not all the upsets in the system lead to erroneous output: some of them are masked by the design. For instance, if the content of a data element changes due to an SEU, but the content is not used in the circuit before it is overwritten with valid data, the SEU does not cause an error in the

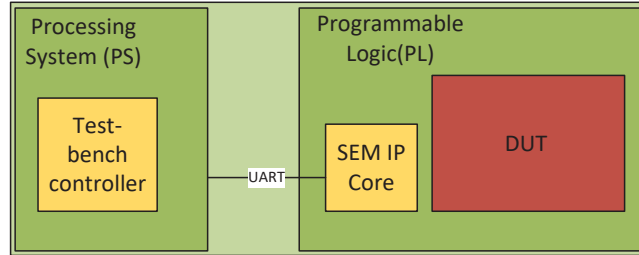


Figure 4: Block diagram of the configuration memory test bench

final output. Moreover, a configuration upset may add a data wire to an unconnected block, effectively leaving the design unchanged. Thus, upset rate estimation without considering the design characteristics and architecture leads to overestimating the amount of errors. The effective cross-section for upsets which impact the results in any way is considered as critical cross-section. This is composed of critical cross-sections for both configuration and data upsets.

5.1. Configuration Memory Critical Cross-section

The configuration bits which directly contribute in design circuits are called *essential bits*. Xilinx tools provide the essential bits. By activating the essential bits report generation in Bitgen tool of Vivado, essential bits are reported in the form of an .ebd file. We are interested in the subset of essential bits named critical bits which impact the final output of the design when upset occurs. Since critical bits are dependent to design functionality and erroneous outputs are required to be defined by the user, Xilinx tools are not able to provide the critical bits.

To obtain the amount of critical bits, we implemented a test-bench for validating the correct design behavior. Figure 4 depicts a block diagram of the test-bench. The test-bench evaluates the correctness of a design behavior while injecting upsets into design configuration bits. SEUs are modeled as bit-flips injected in the configuration bits of the design under test (DUT) implemented in a FPGA. The controller unit implemented on the Processing System (PS) manages the evaluation of critical bits through the following steps:

- Flip essential bits one by one
- For each bit flipped, feed a set of inputs to the DUT
- Compare the output of the DUT with the expected output

The Single Event Mitigation (SEM) IP core provided by Xilinx is used for manipulating the configuration bits. The main function of the SEM IP core is scrubbing for upset detection and correction. In addition to detection and correction, the core can perform SEU injection which performs emulation of SEUs by injecting errors into configuration memory. This fault injection feature

is used in the test-bench to evaluate the SEU effects on design without the need for an expensive radiation test.

Using the SEM IP core, faults are injected to design essential bits one by one. According to the new policy in Xilinx, the essential bits of sub modules of design are not reported in Vivado. Thus the essential bits report generated by Vivado includes the SEM IP essential bits also. In order to extract only the essential bits of the design, we use the *pblock* feature in the floorplaning and mapping steps to force the tool to implement the DUT in specific physical location in FPGA die. Then, the configuration bits corresponding to the physical location are identified using the 7 Series FPGAs Configuration user guide.

FPGA dies are virtually divided to the rows and columns, in which each rows consist of several columns and the width of a column is equal to one CLB. Each column is composed of 36 frames. Frames are smallest addressable part of configuration memory. The configuration bits are organized in set of frames and each frame consist of 101 32-bits word. The physical locations in FPGA die are addressed by the number of the row, column and frame. A specific bit is flipped by providing the SEM IP the injection commands which consist of the physical address : N [*physicaladdress*].

After injecting the SEU to configuration bits, the DUT is activated to generate the output in presence of SEU. The configuration bit is accounted as critical bit, if the generated output is different from the expected one. It should be noted that a critical bit is input-dependent. It means that an infected bit may corrupt the output for some inputs while for others may not. Thus, the experiments are repeated for a set of data inputs. Each input set composed of a set of input vectors that are randomly generated. The fraction of essential bits which affect the circuit functionality are called Design Vulnerability Factor (DVF) and calculated as shown in following:

$$DVF = (E * I)^{-1} \sum_E \sum_I C \quad (2)$$

$$C = \begin{cases} 1, & \text{if generated output} \neq \text{expected output} \\ 0, & \text{otherwise} \end{cases}$$

where E is the total number of essential bits and I is the total number of inputs in the inputs set. The critical cross-section is calculated based on DVF as follows:

$$\text{critical cross-section} = \text{configuration cross-section} * E * DVF. \quad (3)$$

For the critical cross-section analysis, the design under test is an FFT operation, which occupies the majority resources of FPGA among other processing functions of DRS design. The critical cross-section results for this FFT are assumed to be representative of the full design. Table 6 shows the critical cross-section of an FFT implemented in a 7-series FPGA.

Table 6: Determined critical cross-section of an FFT block

Design	essential bits	DVF
FFT	172388	0.0522

Given the results of table 6 only one out of nineteen configuration bits flipped introduce a functional effect in the design. As shown in Section 4.3.3, the incident rate of multi-bit upsets is only 1.7%, meaning the fraction of upsets causing essential configuration bits to flip is approximately equal to the DVF. The resulting critical cross-section for configuration upsets is $5.7 \cdot 10^{-9} cm^2$.

5.2. Data critical cross-section

The critical cross-section for data upsets relates to the amount of data upsets that propagate to design outputs. This heavily depends on the application running on FPGA. A simulation-based fault injection method proposed in [20] is used to estimate the fraction of upsets in user data which propagate through the design and corrupt its output. The SEUs are modeled by a bit-flip injected in user data that lasts for one cycle. Since SEUs in various user data memory cells as well as various clock cycles affect the design differently, the total set of possible SEUs in user data is the multiplication of design latency and total number of user data memory cells used.

In the proposed method, the behavior of design in presence of SEUs is simulated and compared with the correct operation of design while there exists no SEU. To do this, the HDL code of application is modified such that all the user data memory cells are replaced with inject-able ones. In the inject-able version of memory cells, memory cell input is XORed with an inject signal, meaning the inject signal determines whether the memory cell accepts the original input or its flipped counterpart. Table 7 shows the fraction of SEUs in FFT user data memory cells which affect its correct operation in a typical use case. The total set of SEUs injected and the fraction of these injections which leads to wrong operation are presented for flip-flops and BRAMs used, separately.

Table 7: Fraction of SEU in the FFT user data memory cells which affect the correct operation of the FFT processing.

Design		FF	BRAM
FFT	total number of injections	48944	18179
	fraction of injections resulting in an error	0.37	0.28

Assuming the worse case scenario, without simulating the propagation of transient faults in DSP blocks, the resulting critical cross-section for data upsets is $3.4 \cdot 10^{-9} cm^2$. This results in a total critical cross-section of $9.1 \cdot 10^{-9} cm^2$.

5.3. *Semi-critical versus severely critical cross-section*

While non-critical upsets are completely uniform, there is a scale of severity in critical upsets. Most critical configuration upsets that are corrected by the scrubber after several milliseconds might influence several data points, but these errors will likely disappear in averaging the large amount of data points. However, uncorrectable configuration upsets and correctable upsets in certain parts of the control logic will cause the system to enter and stay in an incorrect state.

The semi-critical cross-section can be defined as the cross-section for upsets that result in a minor system error. What classifies as “minor” is unique for each system. For the NCLE DRS, any error which does not interrupt or impede the observation modes and which does not significantly influence the gathered results can be considered a minor error. This translates to all correctable critical configuration upsets in most of the system, with the exception of small control parts. Conversely, a “severe” error is any error which does significantly influence gathered science data. For uncorrectable upsets, this means erroneous data points being accumulated until the device is reconfigured.

For critical upsets in control logic, it is likely that the observation mode run produces some sort of invalid data. It is also possible the run finishes prematurely, never finishes at all, or even overwrites data gathered in previous observation modes stored in flash because of an error in an address calculation. While all of these errors are severe, the latter two are potentially catastrophic. It is unreasonable to assume these catastrophic errors form a significant portion of the severe errors, but the possibility of their occurrence should be taken into account.

Putting a number on the amount of configuration bits that would result in a severe error when upset is difficult, as it is extremely application specific. A reliable way would be extensive testing using fault injection, which is not possible without a semi-final design. For the NCLE DRS design, correctable critical upsets inside of the FFT and filter calculations would not result in a severe system error, and hardware for these calculations make up the grand majority of the area on the FPGA. Because of this, the assumption is made that no more than 25% of the critical upsets result in a severe system error.

For data upsets, a similar problem exists: far from all critical data upsets result in a severe system error, but the fraction is hard to estimate without extensive testing. A small amount of data consists of matters like filter coefficients and loop control variables, which could result in severe propagated errors when upset. Most of the BRAM tiles (which have the highest contribution to the data upset cross-section by far) are used for storing partially processed sample data. Upsets in these bits are guaranteed to never be severely critical. There is not enough insight about the final design available at the time of writing to give a precise estimate on what the severely critical fraction is, but preliminary investigations show that 10% should be a reasonable worst-case figure for the DRS. This includes loop indices, multiplication coefficients and calculated addresses.

Table 8: Expected critical upsets per day (configuration & data)

Situation	Critical upsets	Severely critical
average	0.05	0.01
worst week	5.0	1.0
worst day	23	4.7
worst 5 minutes	87 (0.3/5min)	18 (0.06/5min)

5.4. Calculation of semi- and severely critical cross-sections

The semi-critical cross-section, if defined as a superset of the severely critical cross-section, encompasses all critical configuration upsets ($5.7 \cdot 10^{-9} cm^2$) in addition to all data upsets ($3.4 \cdot 10^{-9} cm^2$). The resulting approximate cross-section is $9.1 \cdot 10^{-9} cm^2$.

The severely critical cross-section encompasses all critical vital configuration upsets and critical uncorrectable upsets ($1.5 \cdot 10^{-9} cm^2$), as well as all critical vital data upsets ($3.4 \cdot 10^{-10} cm^2$), resulting in a total cross-section of approximately $1.9 \cdot 10^{-9} cm^2$. The resulting incident rates can be found in Table 8. While a significant part of these results is based on *very* rough estimates, they can reasonably be considered worst-case numbers.

5.5. Critical undetectable cross-section

The critical cross-section can be further split up into two parts: the detectable and the undetectable critical cross-section. This distinction is useful, as even severe functional problems can largely be mitigated as long as they are detectable, simply by marking the data produced in the rest of the observation mode run as (possibly) invalid. However, for undetectable upsets, this is not an option, meaning they can spoil data without the system being able to mark spoiled data as such.

The undetectable configuration upsets are all MBUs that flip more an even number above three bits in one word. As mentioned in Section 4.3.3, the cross-section for this event is negligible. The data upsets are all assumed undetectable. This means the undetectable cross-section is the same as the critical data upsets for both the semi-critical undetectable cross-section ($3.4 \cdot 10^{-9} cm^2$) and the severely critical undetectable cross-section ($3.4 \cdot 10^{-10} cm^2$).

6. Determining the necessity of additional fault tolerance techniques

The expected errors in the output are only the first piece of information necessary to answer the question whether any additional fault tolerance techniques in the FPGA are necessary for successful completion of the mission. The second is how the application running on the FPGA will respond to these errors, and whether or not these effects can be mitigated without the use of full TMR on the system. Applications relating to imaging or other types of data gathering

are inherently susceptible to noise: the critical (but not severely critical) upsets can simply be considered additional noise in this data. Depending on the application and found frequency of critical faults, post processing back on Earth could mostly or entirely eliminate the effects of these faults.

Considering Table 8, it is clear at least some manner of error mitigation is necessary. While the data conveys a near-worst-case scenario, the estimated number of severe critical upsets is significant: approximately 20 over the total five-year mission duration. This is the amount of times an entire observation mode run is expected to be corrupted or interrupted. As long as none of these events are catastrophic errors which lock up the DRS or destroy significant portions of the measurement data, this should not be enough to compromise any of the science goals. While known to be small, the exact fraction of severe errors causing such catastrophic faults is unknown, so implementing the system without any sort of measures to protect against these events would pose an unacceptable risk. Preferably, measures should also be taken to mitigate the impact of semi-critical events. Upsets that flip the most significant bits of only few samples could have a significant impact on results of long accumulation. With in the worst case approximately one hundred of these events over the full mission duration, this could influence the science data to an unacceptable degree.

6.1. Possibilities for semi-critical upset impact mitigation

As shown in Section 4.3.3, virtually all configuration upsets are detectable and almost all are correctable, but properly recovering from a configuration upset is not trivial. Any data that passed through a struck gate between discovery and the time of the last scrubbing pass can no longer be trusted.

The SEM core has a scrubbing latency of 12.9 ms for the FPGA in the NCLE [18]. To recover, the system would have to either be able to fully roll back to a state from 14ms earlier, or employ (triple) redundant calculations to fall back on in case of a fault. Considering the tight area and power budgets, neither of these is a viable option.

Most observation modes accumulate large amounts of samples into single data points. An option is to store intermediate accumulations (over a time period that is significantly smaller than the expected upset rate) into separate memory locations, as opposed to keeping a single updating data block in flash. The storage budget is fairly large: storing intermediate accumulations for every separate minute would not be prohibitive. By generating metadata for each intermediate accumulation on whether upsets were detected during its collection, faulty intermediates can be filtered out.

It is possible to do this on a more local scale. Due to their accumulating nature, regularly dropping a few milliseconds of data is not a problem in most observation modes. A possibility would be to accumulate results from several milliseconds in an intermediate accumulator, only adding them to the final result if no upsets are detected during that time. The disadvantage is that the data is actually thrown away, instead of marked as possibly faulty.

Using the above approach, approximately twenty times too many data windows would be dropped, as only a small fraction of upsets are critical. The SEM core provides functionality for classifying upsets by checking them to a list of vulnerable bits stored in external storage. This requires the SEM core to interface with the flash memory, and implementing this functionality is a significant amount of work. As shown in Table 2, non-critical configuration upsets have an expected occurrence rate below once a minute during the worst five minutes of the mission, meaning dropping several milliseconds of data is not a large problem. Due to the short lead-time of the project and low priority of this feature, this option was not considered viable for the NCLE DRS.

For observation modes where a high temporal resolution (order of milliseconds) is necessary, the previous approach is not an option. These modes can not run for much longer than a few seconds, as the bandwidth to get that much data back to Earth is not available. A realistic alternative is to simply assume the system does not experience any upsets in relevant bits during that time. Even during the worst possible hour of the mission, the average time to a critical upset is approximately 25 minutes; significantly longer than these measurements take.

As with storing intermediate accumulations, each run can be marked with metadata on whether upsets were detected during the collection of the data. Even data collected while the system was influenced by an upset is likely valuable to some extent (especially since most upsets are non-critical) and the decision to use it can be made on Earth.

6.2. Severely critical upset mitigation

Due to the extremely low incident rate, an acceptable response to severely critical upsets in the NCLE DRS would be to drop all tasks and request a power cycle from the CDHS. However, classifying the severity of detected upsets possibly poses a problem and not all severely critical upsets are detectable in the first place.

It is possible to decrease the incident rate of severe incidents by using traditional fault-tolerance techniques such as TMR. With the incident rates of critical faults, getting simultaneous faults in two voters can be considered statistically impossible, as long as they are not placed adjacently. The small area and power budgets are less of a problem in this case, as the severely critical sections use only a small fraction of both of these budgets. Implementing TMR in the severely critical sections would eradicate the grand majority of the severely critical faults from the system, both those caused by configuration upsets as well as those caused by data upsets.

6.3. Catastrophic event mitigation

As catastrophic events are mostly comprised of a small subset of severely critical upsets, selectively applied TMR would drive down incident rate significantly. While the catastrophic cross-section should be close to negligible, some

possibility for catastrophic errors still remains, mostly in the Kintex-7 system bits that cause full functional interruption when upset.

The cross-section for functional interruption events in the Kintex-7 is significantly smaller than the cross-section for latch-ups [7], which was considered negligible in section 4.1. However, due to the catastrophic nature of these events, some sort of mitigation is in order. An option is establishing a “heartbeat” line to the CDHS, which can reconfigure and power cycle the FPGA, to automatically trigger a reset whenever this heartbeat line goes flat. To offer further protection, this heartbeat could carry additional information about the DRS system state: if an observation mode run takes longer than it should, CDHS could respond similar to the heartbeat going flat.

6.4. Overhead and profit of fault tolerance features used in the NCLE DRS

The SEE mitigation used in NCLE mainly relies on the Xilinx SEM IP for scrubbing and upset detection and FPGA health monitoring from the external CDHS. Observation data frames acquired during a detected SEU are labeled as such in the data storage to enable filtering of suspect data during post-processing. The usage of full TMR was considered, but was discarded after results from the presented analysis became clear. Replicating parts of the data processing infrastructure was deemed prohibitively costly in both resource requirements and power consumption, both of which would become three times as large. The performance deterioration resulting from downsizing the design was deemed more severe than the expected performance deterioration resulting from SEEs in these parts. As such, only the observation mode control parts were deemed candidates for TMR. In the end, these parts were analyzed to have only a very limited critical cross-section and are expected to perform sufficiently without TMR.

As mentioned, the Xilinx SEM IP is used for scrubbing the configuration memory upsets and detecting potential problems. Since the scrubber is itself implemented on the FPGA, it introduces some overhead in FPGA resource utilization. This is shown in Figure 5. The SEM IP was configured to apply error correction with repair mode selected (basic scrubbing) at 40MHz, the running frequency of the observation controller module. Error classification was not activated but the monitor port of the SEM IP block was connected to the observation controller. This allows for detected SEUs to be logged into the metadata for the observations. As is shown in Figure 5, the SEM IP overhead is only a small fraction of the overall FPGA resources: less than 1% each for block-RAMs (BRAM), flip-flops (FF), lookup-tables (LUT), and lookup-table RAMs (LUTRAM), and close to 3% for the global routing buffers (BUFG) which are mainly used for clock routing and are partially shared with the FPGA logic implementing the observation modes.

Finally, one key element in the DRS system that did require protection was the storage of observation data. This storage contains the pre-processed sample data and its related metadata, and is stored in NAND-flash memory on the FPGA board. To prevent accidental corruption of the data a combination

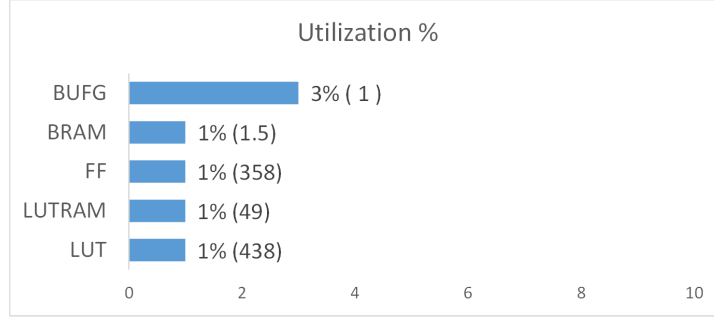


Figure 5: Scrubber resource utilization % (exact resource usage) obtained from Vivado post-implementation report

of ECC memory techniques with scrubbing, CRC checks on individual data blocks, and automatic bad block management were used. This ensures that, once recorded, the data can safely be transferred for analysis and possible transfer errors can easily be detected.

7. Conclusion

It has been shown reasonable to expect a significant amount of upsets in the NCLE DRS during the mission, but few are expected to be critical, and even fewer are expected to possibly compromise the mission goals. When using the built-in scrubber of the Kintex-7 as the only radiation error mitigation, there would be a small but non-zero risk of catastrophic events in the FPGA endangering the acquisition of valid science data for the NCLE mission.

The most troubling risks were addressed with simple, realistic, low-overhead mitigation options. Given the proposed mitigation strategy for the NCLE mission, radiation related errors in the digital receiver system are extremely unlikely to compromise the science goals of the NCLE mission.

An approach has been presented to analyse the expected radiation-related system faults in FPGA devices when used in space missions. In addition by running proposed test-benches, the percentage of these faults which affect the final output of DRS were calculated. Experimental results show that only 5%, 37% and 28% of the faults in configuration memory, flip flops and BRAMs respectively corrupt final results of DRS. However, the presented failure rate predictions also show that the probability of SEE causing problems in the actual observation data is very low for most of the operational life of the instrument. These results allowed the fault mitigation strategy in the NCLE DRS to focus mainly on fault detection and logging, and less on correction, saving a significant amount of power.

Acknowledgements

The NCLE scientific payload development is supported by ESA PRODEX and The Netherlands Space Office (NSO). The Netherlands-China Low-frequency Explorer (NCLE) instrument was designed and built in the Netherlands by a team consisting of scientists and engineers from the Radboud Radio Lab (RLL) of the Radboud University (Nijmegen), the Dutch institute for radio astronomy ASTRON (Dwingeloo) and Innovative Solutions In Space (ISIS, Delft). The authors would like to thank their collaborators in the NCLE team, as well as the other members of the Radboud Radio Lab, for their helpful input at various stages of the project. The work presented in this paper was also supported by the ITEA3 project 14014 ASSUME.

References

- [1] S. Jester, H. Falcke, Science with a lunar low-frequency array: from the dark ages of the universe to nearby exoplanets, *New Astronomy Reviews* 53 (1-2) (2009) 1–26. doi:10.1016/j.newar.2009.02.001.
- [2] P. Zarka, B. Cecconi, W. Kurth, Jupiter’s low-frequency radio spectrum from Cassini/Radio and Plasma Wave Science (RPWS) absolute flux density measurements, *Journal of Geophysical Research: Space Physics* 109 (A9). doi:10.1029/2003JA010260.
- [3] M. Klein Wolt, A. Aminaei, P. Zarka, J.-R. Schrader, A.-J. Boonstra, H. Falcke, Radio astronomy with the European Lunar Lander: Opening up the last unexplored frequency regime, *Planetary and Space Science* 74 (1) (2012) 167–178. doi:10.1016/j.pss.2012.09.004.
- [4] Xilinx, Device reliability report, first half 2016 (UG116) (Dec. 2016).
- [5] D. Hiemstra, V. Kirischian, Single event upset characterization of the Kintex-7 field programmable gate array using proton irradiation, in: *Radiation Effects Data Workshop (REDW)*, 2014, pp. 1–4. doi:10.1109/REDW.2014.7004593.
- [6] D. Lee, M. Wirthlin, G. Swift, A. Le, Single-event characterization of the 28 nm Xilinx Kintex-7 field-programmable gate array under heavy ion irradiation, in: *Proceedings of the Radiation Effects Data Workshop (REDW)*, 2014, pp. 1–5. doi:10.1109/REDW.2014.7004595.
- [7] V.-M. Plăcintă, L.-N. Cojocariu, O.-E. Huțanu, F. Maciuc, M. Straticiuc, C.-A. Tănase, L. Arnold, S. Wotton, Kintex-7 irradiation, test bench and results, in: *TWEPP 2016 - Topical Workshop on Electronics for Particle Physics*, 2016.
- [8] ESA-ESTEC, Space product assurance: Techniques for radiation effects mitigation in ASICs and FPGAs handbook, Tech. Rep. ECSS-Q-HB-60-02A, ESA Requirements and Standards Division (Sep. 2016).

- [9] L. van Harten, R. Jordans, H. Pourshaghagh, Necessity of fault tolerance techniques in Xilinx Kintex-7 FPGA devices for space missions: A case study, in: Proceedings of the 20th Euromicro Conference on Digital System Design, 2017, pp. 299–306. doi:10.1109/DSD.2017.45.
- [10] ESA-ESTEC, Engineering standard: Space environment, Tech. Rep. ECSS-E-ST-10-04C, ESA Requirements and Standards Division (Nov. 2008).
- [11] D. Heynderickx, B. Quaghebeur, H. D. R. Evans, The ESA space environment information system (SPENVIS), COSPAR Scientific Assembly 34, online tool interface available at <https://spenvis.oma.be/>.
- [12] S. Seltzer, Updated calculations for routine space-shielding radiation dose estimates: SHIELDOSE2, NISTIR.
- [13] J. M. Armani, J. L. Leray, V. Iluta, TID response of various field programmable gate arrays and memory devices, in: Radiation and Its Effects on Components and Systems (RADECS), 2015 15th European Conference on, IEEE, 2015, pp. 1–4. doi:10.1109/RADECS.2015.7365686.
- [14] Lattice, LatticeECP3 family data sheet (Mar. 2010).
- [15] N. S. Saks, M. G. Ancona, J. Modolo, Generation of interface states by ionizing radiation in very thin MOS oxides, IEEE Transactions on Nuclear Science NS-33 (6) (1986) 1185–1190. doi:10.1109/TNS.1986.4334576.
- [16] H. Takai, M. Wirthlin, A. Harding, Soft error rate estimations of the Kintex-7 FPGA within the ATLAS liquid argon (LAr) calorimeter, Journal of Instrumentation 9 (2014) 1–8. doi:10.1088/1748-0221/9/01/C01025.
- [17] P. O’Neil, G. D. Badhwar, W. Culpepper, Risk assessment for heavy ions of parts tested with protons, IEEE Transactions on Nuclear Science 44 (6) (1997) 2311–2314. doi:10.1109/23.659052.
- [18] Xilinx, Soft error mitigation controller v4.1 LogiCORE IP product guide (Apr. 2017).
- [19] M. Wirthin, D. Lee, G. Swift, H. Quinn, A method and case study on identifying physically adjacent multiple-cell upsets using 28-nm, interleaved and SECDED-protected arrays, IEEE Transactions on nuclear science 61 (6) (2014) 3080–3087. doi:10.1109/TNS.2014.2366913.
- [20] M. Mousavi, H. Pourshaghagh, M. Tahghighi, R. Jordans, H. Corporaal, A generic methodology to compute design sensitivity to SEU in SRAM-based FPGA, in: Proceedings of the 21st Euromicro Conference on Digital System Design, 2018, pp. xxx–yyy, accepted.



Louis Daniël van Harten is a final-year MSc student in the Electrical Engineering program of the Eindhoven University of Technology. He joined the NCLE team as part of an internship, being involved with the fault-tolerance aspect of the Digital Receiver System. He is currently working on the design and implementation of an interferometry-based high-accuracy tracking system for rockets as part of the Eindhoven University of Technology and Radboud University Nijmegen combined student team PR³, creating a payload for the REXUS/BEXUS program with the Swedish National Space Board (SNSB) and Deutsche Zentrum für Luft- und Raumfahrt (DLR), in collaboration with ESA.

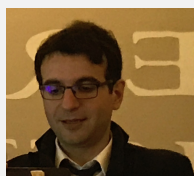


Mahsa Mousavi received the B.S. and M.S degree in computer engineering from the Isfahan University of Technology, Isfahan, Iran, in 2008 and 2011, respectively. She received the MPhil degree in electronic and computer engineering from the Hong Kong University of Science and Technology, Hong Kong in 2017. She is currently pursuing the PhD degree in electrical engineering at Eindhoven University of Technology, Netherlands. Her research interests include Embedded systems, FPGAs, VLSI designs and fault tolerance systems. She is currently working on FPGA-based fault tolerance design and in parallel collaborate with NCLE team in the Radboud University Nijmegen on the fault-tolerance aspect of the Digital Receiver System.



Roel Jordans received both the MSc and PhD degrees in field of Electrical Engineering from Eindhoven University of Technology in 2009 and 2015 respectively. He worked within the PreMaDoNA project on the MAMPS tool flow as a researcher afterwards. His dissertation focussed on the automatic design space exploration of VLIW ASIP instruction-set architecture within the ASAM project. His research interest include compilers and compilation techniques for application specific systems, digital signal processing systems based on customized VLIW architectures,

and reliability and fault-tolerant design for space applications. Currently he is an Assistant Professor at the Eindhoven University of Technology where he teaches parallelization, compilation, and heterogeneous data processing platform architectures. In parallel he is employed at the Radboud University Nijmegen where he is active as science DSP architect in the Radboud Radio Lab. He also serves as a program committee member for the EUROMICRO Symposium on Digital System Design.



Hamid Reza Pourshaghghi received his PhD in the Electronic Systems group at Eindhoven University of Technology (TU/e) in the Netherlands, in 2013. His research was funded by the Dutch Technology Foundation STW on the project “Synthesis and Implementation of Adaptive Energy-Aware Circuits and Systems”. His research was partly done in cooperation with NXP Semiconductors in Eindhoven (2011-2012).

From 2013 to 2015, Dr. Pourshaghghi had been working as a post-doctoral researcher fellow within the astrophysics department of Radboud University Nijmegen in the Netherlands. He mainly focused on the design of DSP digital receivers for radio astronomy science in space. He is currently the lead engineer at Radboud Radio Lab (RRL) at Radboud University Nijmegen. He is also the Project Manager for the design and construction of Digital Receiver System for NCLE Moon mission. Since April 2016, he has been working as an Assistant Professor at the Electronic Systems group at TU/e.